

12.Component Diagram

Contents

12.1	Pendahuluan.....	2
12.2	Implementasi Perangkat Lunak	3
12.2.1	Software Tools	3
12.3	Component Diagram.....	3
12.3.1	Component	3
	Referensi.....	8

12.1 Pendahuluan

Klasifikasi diagram UML:

- **Static**
 - Use case diagram, Class diagram
- **Dynamic**
 - State diagram, Activity diagram, Sequence diagram, Collaboration diagram
- **Implementation**
 - Component diagram, Deployment diagram

Klasifikasi lain dari UML

- **Behavior diagrams**

Jenis diagram yang menggambarkan perilaku suatu sistem

- Meliputi activity, state machine, dan use case diagrams, interaction diagrams

- **Interaction diagrams**

Sebuah subset dari diagram perilaku yang menekankan interaksi objek. Meliputi collaboration, activity, sequence diagrams

- **Structure diagrams**

Jenis diagram yang menggambarkan unsur-unsur spesifikasi yang terlepas dari waktu. Meliputi class, composite structure, component, deployment

UML component diagrams : mendeskripsikan komponen software dan kebergantungannya dengan yang lain. Komponen merupakan unit otonom dalam sistem. Komponen dapat digunakan untuk mendefinisikan ukuran dan kompleksitas sistem S/W. komponen diagram UML memungkinkan untuk memodelkan komponen perangkat lunak tingkat tinggi, dan interface untuk komponen tersebut. Komponen dan subsistem dapat di-REUSED dan di-REPLACED.

Terdapat Kebergantungan antara 2 elemen, jika terdapat perubahan pada 1 elemen bisa mempengaruhi lainnya. Diagram komponen sering disebut sebagai "wiring diagrams". Wiring komponen merepresentasikan komponen dan dependensi diantara komponen tersebut.

12.2 Implementasi Perangkat Lunak

12.2.1 Software Tools

Implementasi perangkat lunak memerlukan beberapa tools yang digunakan untuk mengembangkan sistem. Untuk memastikan bahwa software yang digunakan versinya kompatibel, lisensi banyaknya pengguna dan sebagainya maka diperlukan adanya aturan manajemen. Software tools yang digunakan antara lain

- a. Software modelling Tools
- b. Compiler, Debugger, Interpreter, runtime support
- c. Visual editor
- d. Integrated Development environment
- e. Configuration Management
- f. Class Browser
- g. DBMS
- h. CORBA
- i. Application Container
- j. Testing Tools
- k. Installation Tools
- l. Conversion Tools
- m. Documentation Generator

12.3 Component Diagram

12.3.1 Component

Sebuah komponen digambarkan dengan segiempat dimana pada bagian atas diberi label component. Pada bagian kanan atas terdapat icon component yang dapat ditampilkan ataupun tidak ditampilkan. Label bisa diberi nama dengan menggunakan stereotypes standart antara lain : <<subsystem>> atau <<entity>>.

Component bisa memiliki beberapa bagian antara lain :

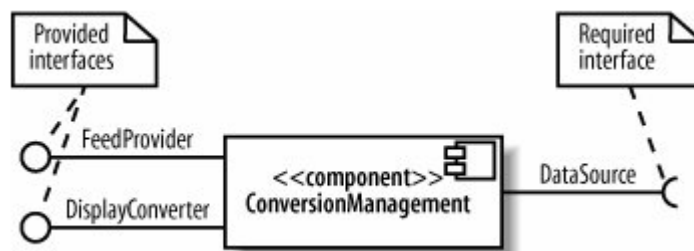
- Interfaces
Sebuah interface merupakan deklarasi dari satu set operasi dan obligations
- Usage dependencies
usage dependency adalah hubungan yang salah satu unsur memerlukan unsur lain untuk implementasi penuh
- Ports
Port merupakan titik interaksi antara komponen serta lingkungan
- Connectors
Menghubungkan 2 buah components

Interface

- Merupakan definisi dari kumpulan satu atau lebih operasi
- Hanya menyediakan operasi tetapi tidak implementasinya
- Implementasi biasanya disediakan oleh kelas / komponen
- Dalam sistem yang kompleks, Implementasi disediakan oleh sekelompok kelas daripada satu kelas

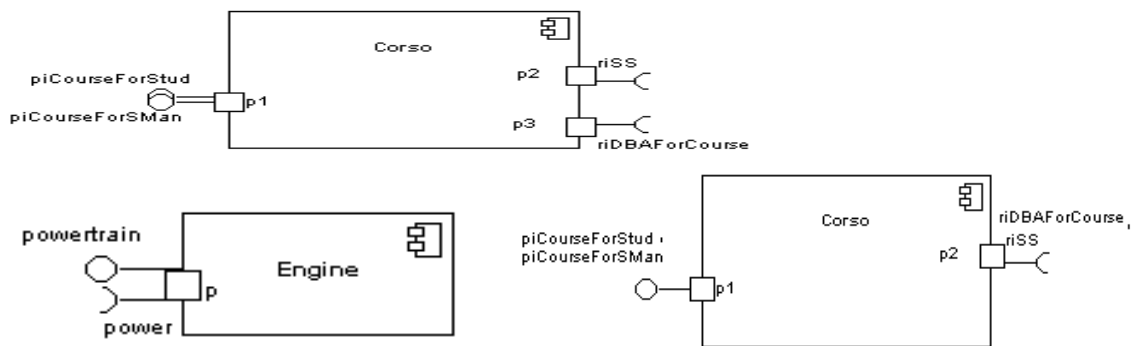
Ada 2 jenis interface, yaitu :

- Provide interface
Antar muka untuk menunjukkan bahwa component menyediakan suatu fungsi
- Required Interface
Antar muka untuk menunjukkan bahwa component memerlukan suatu fungsi



Port

- Semua interaksi komponen dengan lingkungan dilakukan melalui port.
- Internal tertutup rapat dari lingkungan
- Tidak terdapat pada UML 1.x



Dependencies

- dimodelkan dengan garis terputus dengan panah terbuka $\text{-----}\Rightarrow$
- gambarkan dependencies dari kiri ke kanan
- Contoh:

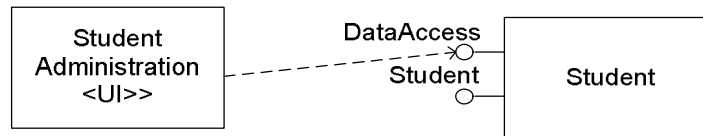
<<ASP>> Source Code bergantung pada

<<database>> MySQL



- Dimungkinkan sebuah component dependencies pada interfaces component lainnya

Contoh:



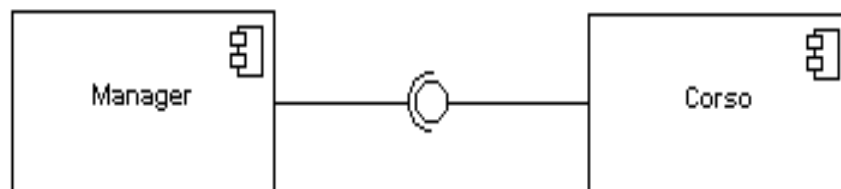
Connector

Terdapat 2 jenis connector yaitu :

- Assembly
- Delegation

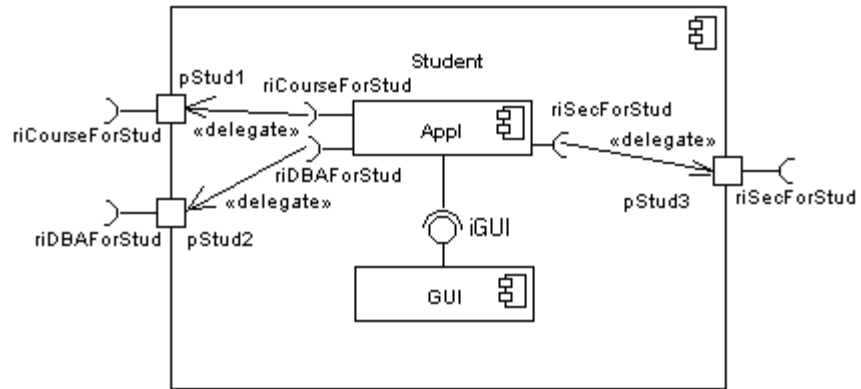
Assembly Connector

- Merupakan konektor diantara 2 komponen mendefinisikan bahwa satu komponen memberikan layanan yang dibutuhkan oleh komponen lain.
- Harus didefinisikan dari required interface ke provided interface
- Dinotasikan dengan "ball-and-socket" connection



Delegation Connector

- Merupakan link antara external component ke internal realization
- Representasi dari signal



Referensi

1. OReilly.Learning.UML.2.0.Apr.2006
2. Simon Bennet, Steve McRobb and Ray Farmer, *Object Oriented Systems Analysis and Design Using UML*, Edisi 3. ; McGraw Hill, 2006. (SB)